From K-Anonymity to Differential Privacy Back to K-Anonymity!

Ddeel ANJUM, Guillame Raschia, Marc Gelgon
LINA University of Nantes
France
adeelanjum2001@hotmail.com

ABSTRACT: *In this paper, we address the K-anonymity issue. For preserving privacy in data publishing, K-anonymity algorithms and semantic privacy models have potential. We proved this fact with adequate discussions and illustrations. We hope to document further the K-anonymity issue in the forthcoming research.*

Keywords: Privacy preservation, data security, K-anonymity

Received: 1 August 2014, Revised 8 September 2014, Accepted 14 September 2014

1. Introduction

The research community has left no stone unturned in devising strategies for both syntactic and semantic privacy definitions. The literature on privacy protection reveals that no privacy model is capable of incorporating growing demands of data publication (e.g., the adversarial background, needs of data publisher, constraints on underlying dataset etc.). While syntactic privacy definitions, being widely studied for PPDP task, requires assumptions that make them questionable w.r.t privacy guarantees in critical applications, each syntactic approach is based on an attack model of an adversary and it assumes that such an adversarial knowledge is limited and is pre-defined. Consequently, these approaches fail to provide the promised degree of protection if the adversarial knowledge exceeds the protection level provided by the given privacy model. In short, it is difficult and impossible to model the adversarial background knowledge. Semantic privacy definition e.g., differential privacy (DP), was introduced to overcome the inherent deficiencies in syntactic privacy approaches but its applicability in real life situation is questioned frequently. Thus, despite these countless efforts, privacy protection remains an open issue.

2. Achieving K-anonymity

2.1 Warm-up

A database satisfies K-anonymity if every record is indistinguishable on quasi identifiers from at least K - 1 other records. This indistinguishability principle [18] supports an equivalence relationship on the records of an anonymous public release and

prevents from identity disclosure of individuals with a probability of 1/*K*.

More precisely, let *D* be a database where each element *x* is a point in a *k*-dimensional space $\mathbb{N}^k$. In the usual *K*-anonymity problem statement, there are 3 bins of attributes, resp. the identifiers (*id*), the quasi identifiers (*qi*) and the sensitive attribute (s). A *K*- anonymous release of a micro-dataset *D* is mainly achieved b.t.w. of a sanitization mechanism that blur the distinctness of records within groups of at least *K* records each. The sanitized dataset is computed by generalizing records safely.

Definition 2.1.1 (Generalization mechanism A)
Given a database schema *D* (*id*, *qi*, *s*), a generalization mechanism is a bijective function *A* defined as follows:

$A: D \longrightarrow D$
$D \longmapsto A(D) = \{(t[id], v, t[s]) \mid t[qi] \leq v \wedge t \in D\}$

where *v* is a generalized value of *t* [*qi*] according to any pre-dened partial order over *dom* (*qi*). It is worth to notice that *A*(*D*) is not unique since there exist many different ways to generalize *t* [*qi*] and the *A*(*D*) enumeration is properly combinatorics. Then, regular approaches try to optimize a utility-based objective function in the generalization mechanism. This is the underlying reason why the generalization mechanisms for *K-anonymization* have been proved to be *NP-hard* [10].

Many approximation algorithms have been proposed in the literature since the seminal work of *L*. Sweeney [17]. Usually, Mondrian approach [11] is thought of as the baseline algorithm since it has the basic good properties we could expect from such algorithms: local recoding and multi-dimensional partitioning. Mondrian iteratively operates a binary partitioning of the data space until every block contains between *K* and 2*K* −1 points. Actually, Mondrian builds a kd-tree over the raw data and publishes bounding boxes of the leaves as equivalence classes of the anonymous release. Construction has time complexity *O* (*n* . *log n*), where $n = |D|$ is the number of records in raw data.

2.2 Scalable Yet Useful K-Anonymity Mechanisms
Space partitioning by the way of Point Access Methods (PAM) is a key concept for achieving K- anonymity. We briefly review here two of the most appealing approaches and present their features.

2.2.1 R^+– Tree-Based Mechanism
Following the geometric representation of the data, Iwuchukwu et al. [9] propose to use a bulk-loading implementation of an R+-tree, one of the most popular spatial access methods for databases, to compute the K-anonymous release. It outperforms Mondrian thanks to buffering and efficient bottom-up index construction algorithm, and it scales up to very large data sets. Furthermore, the hierarchical structure of the R+- tree natively supports (K^{lB})-anonymity for all level l in the tree, with *B* the fanout parameter. And with an ordered leaf scan, it could support (*pK*)- anonymity as well, for all *p* in $\mathbb{N}$. Time complexity remains in *O* (*n*. *log n*) And I/O cost for external computation is in *O* (*n* /*B*.*log n* = *B*).

Since the R+-tree bulk-loading algorithm is applied on a set of points rather than a set of spatial objects with an extent, it is actually a variant of a *kd-B-tree* structure where hyper-rectangles have been shrinked to the minimum bounding boxes (MBB) of the subset of points in each equivalence class. Remind that a *kd- B-tree* is a bucket-oriented variant of a *kd-tree* where the fanout of each node is defined by parameter *B* that usually ts the disk block size.

2.2.2 BANG le-based mechanism
Despite historical proximity with the Grid le and its DYOP variant, directory of the BANG le is a tree rather than an array (grid). It follows H. Samet's claim [16] who states that the BANG le is a variant of the *kd-B-tree*, that is a kd-B-tree with regular decomposition. Blocks are in the leaves whereas inner nodes contain entries of the form (subspace spanned by a child node, reference to child node). The subspace spanned by a child node is denied as an outer most hyper-rectangle and zero or more nested regions to remove.

The partitioning algorithm is simple yet efficient. As any kd-B-tree, its time complexity is *O*(*n* . *log n*). For external storage, I/O cost still remains in *O* (*n*/*B log n*/*B*) with *B* the disk block size. It performs iterative insertion of data points in a top-down manner. Enclosing grid region identifier is first encoded thanks to a *Z*-ordering-like scheme. The all path up to the root (the entire space) is also retrieved. Then, the BANG directory is searched for the smallest recorded region that encloses the data point. It is then

trees partitioning is performed and privacy budget is allocated in a geometrically increasing way to counts during the partitioning of 2D data. An ordinary least squares (OLS) estimator is devised to achieve consistency and minimum error variance in time linear in the size of the published tree. Quad-tree partitioning is found to be fast and superior in quality of the output to any other method. assigned to the proper bucket.

When data bucket overflows, the algorithm operates splitting to balance the distribution of points between buckets. Splitting is done by iteratively halving the space spanned by points in the bucket until the best balance is achieved. It gives birth either to a buddy region or to a new enclosed region. The iterative halving strongly differs from the R+-tree splitting strategy. Indeed, the BANG le operates from the entire space to blocks (top-down) whereas the R+-tree operates from points to blocks (bottom-up). This distinct feature has significant impact on performance.

Finally, while preserving the many good features of any PAM approach, the BANG Anonymization mechanism [1] outperforms its R+-tree counter-part both in utility and scalability.

2.3 Known limitations

Syntactic privacy definitions, such like K-anonymity and the many derived privacy models (*l*-diversity, *t*-closeness, etc.) being widely studied for PPDP task, requires assumptions that make them questionable w.r.t. privacy guarantees in critical applications. Each syntactic approach is based on an attack model of an adversary and it assumes that such an adversarial knowledge is limited and is predefined. Consequently, these approaches fail to provide the promised degree of protection if the adversarial knowledge exceeds the protection level provided by the given privacy model. In short, it is difficult to impossible to model the adversarial background knowledge. Semantic privacy definitions e.g., differential privacy DP, was introduced to overcome the inherent deficiencies in syntactic privacy approaches.

3. Achieving Data Privacy with Semantic Guarantees

Roughly speaking, going from syntactic privacy models to semantic models requires to incorporate randomization process within the mechanism and to define constraint on the process itself rather than its output. Differential privacy [3] is the most popular semantic privacy model and it has been widely studied from the past seven years.

3.1 Preliminaries

Definition 3.1.1 (ε, δ)-differential privacy (DP) Given a randomization mechanism $M : D \longrightarrow R$, we say that M satifises (ε, δ)-dp if for all $D, D' \in D$ such that $|D \Delta D'| = 1$ and for all $R \subseteq R$, the following inequality holds:

$$P[M(D) \in R \leq e^{\varepsilon} P[M(D') \in R] + \delta$$

The two main randomization mechanisms M performed to achieve *DP* are the Laplace [3] and the Exponential [14] mechanisms. The former is used when the output is numerical while the latter is for discrete outputs or when it is meaningless to introduce scalar noise. Other mechanisms are Li et al's matrix mechanism [12], the geometric mechanism (a discretized version of the Laplace mechanism) by Ghosh et al. [8] and the Gaussian mechanism [5].

Achieving DP in a PPDP task i.e., non interactively, essentially yields to "*privately*" answering linear queries, and especially counting queries over the set of records. After private histogram computation, there have been many proposals to "*privately*" answer any range queries. To this end, a recent line of research focused on partitioning techniques to support randomization mechanisms.

3.2 Data partitioning for DP

As usual, a popular approach to partitioning relies on kd-trees: at each level, an attribute is chosen and points in the database are split in 2 disjoint sets according to some criteria. Usually uniformity in the number of points on both sides of the splitting line is considered by choosing the median. Noisy counts of the two newly founded partitions are then published and partitioning is done recursively.

The idea of differentially private data partitioning index structures is suggested in the context of private record matching in [4]. The approach there relies on an approximate mean as a surrogate for the median (on numerical data) to build kd-trees. The approach of Xiao et al. [21] imposes a fixed resolution grid over the micro-data. It then builds a kd-tree based on noisy counts in

the grid, splitting nodes which are not considered '*uniform*', and then populates the final leaves with '*fresh*' noisy estimated counts. Quad-tree partitioning simply imposes a recursive fixed grid in which at each step the space is divided into four rectangular cells of the same size. In [2] by Cormode et al., a comparison between several median finding methods, Hilbert R-trees and quad-trees partitioning is performed and privacy budget is allocated in a geometrically increasing way to counts during the partitioning of 2D data. An ordinary least squares (OLS) estimator is devised to achieve consistency and minimum error variance in time linear in the size of the published tree. Quad-tree partitioning is found to be fast and superior in quality of the output to any other method.

4. Relaxing Semantic Privacy Definitions for Syntactic Approaches

There is a very recent trend of relaxing DP so that both syntactic and semantic privacy approaches can ourish together in order to remove each others deficiencies. Gehrke et al. [7] proposed to exploit the adversary's uncertainty about the underlying dataset. The authors of [7] stated that adding a random sampling preprocessing step provides a natural way in capturing the adversarial uncertainty about the input dataset. Consequently, they initiated a new privacy definition coined Crowd Blending Privacy that permits to design new mechanisms having better applicability regarding utility/efficiency than differentially private mechanisms while keeping the notion of privacy intact.

Definition 4.0.1 Crowd Blending Privacy [7] A sanitization mechanism M satifies crowd blending privacy if for every dataset D and every individual $i \in D$, either i ε-blends in a crowd of K people in D w.r.t. M, or $M(D) \approx_{\varepsilon} M(D \setminus \{i\})$ (or both).

Crowd blending privacy compels the mechanisms either to blend an individual i in a group of K individuals or do not release the data at all. This way the mechanisms actually do not release any information about i, apart from the general properties of the crowd of K individuals.

Moreover, the authors of [7] force these mechanisms to satisfy the crowd blending privacy in pursuance of achieving differential privacy (and even zero-knowledge privacy) when the underlying dataset is randomly sampled from a given population. Actually, they prove that DP implies crowd blending privacy i.e., removing the condition $M(D) \approx_{\varepsilon} M(D \setminus \{i\})$ from the denition of crowd blending privacy, still results in *DP*.

Li et al. [13] in an open manuscript, have previously introduced the concept of "*safe*" *K- anonymity* and argued that safe K-anonymity preceded by a random sampling step satises (δ, ε)-DP. The authors proposed a relaxed differential privacy denfintion under sampling:

Definition 4.0.2 $(\beta, \varepsilon, \delta)$-DP [13] Given a dataset D, a sanitization mechanism A satisfies $(\beta, \varepsilon, \delta)$-DP *iff* $\beta > \delta$ and a mechanism $A\beta$ satisfies (ε, δ)-DP such $A\beta$ that $A\beta$ samples records from D with a probability β.

Gehrke et al. noticed that K-anonymity is based on the premonition of "*blending in a crowd*", since the records in a *K-anonymous* sanitized release are required to "*blend*" with at least $K-1$ other records. Ostensibly, the idea of blending in a crowd of many people is sufficient to protect the privacy of concerned individuals. However, as shown by several known attacks, K-anonymity is unable to fully capture this notion of "*blending in a crowd*", because it does not impose any constraint on the mechanisms used to provide the *K-anonymous* release.

One of the important directions given by the authors of [7] is the adaptation of generalization based *K-anonymity* solution to DP. They maintain that if generalization is not done carefully, the privacy of individuals is at risk. However, they show if the generalization step is performed gingerly, these generalization-based *K-anonymity* algorithms can satisfy crowd blending privacy.

This interesting trend of combining DP with generalization-based approaches has given the opportunity to blend the strength of DP with the efficiency of state-of-the-art generalization algorithms for practical privacy. Though this line of research is in the initial stages, we may in the near future benefit from both research tracks. Promoting and contributing to this line of research is essentially the purpose of this communication.

5. Point Access Methods for Privacy Preservation: One Step Further

In this section, we re-investigate the combination of multi-dimensional partitioning techniques with semantic privacy models. More precisely, we give an overview of two distinct tracks: we argue why the first one looks like a dead-end, whereas the other one seems to be relevant and by far more impacting on practical PPDP tasks.

5.1 Improving DP Partitioning on a Straight Line

The possibility of improving existing partitioning strategies was investigated. In particular, we noticed how the works of Xiao et al. [20] and Cormode et al [2] adopt quite basic schemes (kd-trees for the former and both kd-trees and quadtrees for the latter). Cormode suggests the quad-tree is the best partitioning of the several methods tested. Even if the quadtree is the simplest one, we were wondering how it would perform on high dimensional space, for which more elaborated existing partitioning schemes with well studied properties are known. In particular, we focused on the PK-Tree [19] and on the BANG le [6].

We will limit the exposition to the PK-Tree as it is the most straightforward to understand while giving some insights into issues we had about partitionings in general. The PK-Tree can be any partitioning scheme which satisfies the requirements of regular decomposition, containment and mutual disjonction, plus a fourth one, peculiar to the PK-tree. It mandates that each node v except the root must have at least k children, which can be either points or regions. Moreover, if there are more than k children there must be no region in the regular decomposition which can contain at least *k* of these children. Nodes satisfying this condition are called k-instantiable. An illustration of a pk-tree based on an underlying quad tree implementation is provided in Figure 1. In the picture we can see each node has at least *k* children, which can be a mix of both points and nodes. If all the requirements are met then the PK-tree exhibits inexpensive updates, efficient storage and a bound on the maximum number of children per node. Average height is proven to be bounded for some classes of data. Another interesting property of the PK-tree is that it is unique, that is, if the underlying decomposition is

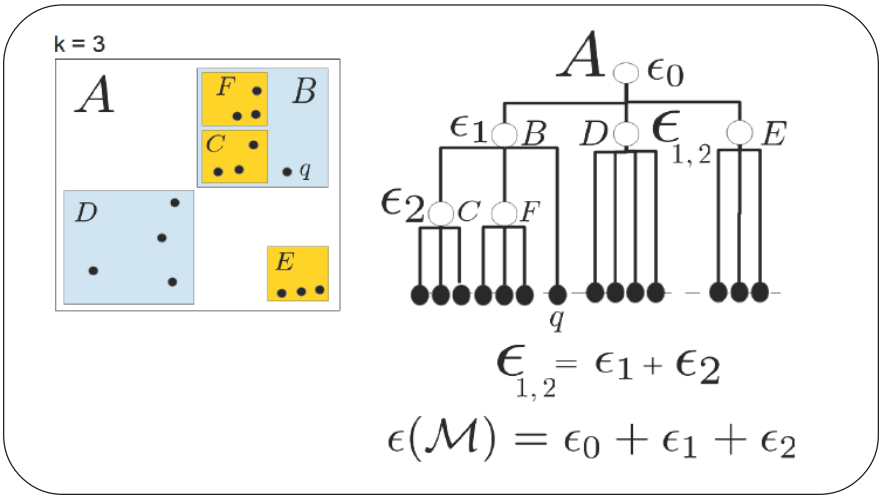

Figure 1: PK-Tree partitioning with privacy cost

regular (like i.e. quadtree) after a series of insertions and deletions the last step, regardless of the order in which they were inserted.

One possible algorithm to make the PK-Tree differentially private could be made in two steps. First we could take the noisy counts of a ne grained regular grid of the space, and then we could use this approximate knowledge about points distribution to drive the PK partitioning for taking new fresh noisy counts from the database. This kind of two-step procedure is adopted in Xiao [20] for kd-trees. Since the tree would be unbalanced we could increase the " budget reserved to leaves that are not at maximum depth: for example, in Figure 1 nodes D and E have a reserved budget of $\varepsilon_1 + \varepsilon_2$ because the partitioning procedure decided it was unnecessary to further subdivide the regions. After the partitioning, it would be important to *x* the noisy counts in order to make them consistent: the counts of children nodes should sum up to the counts of their parents. Ideally, this manipulation should also aim at reducing the variance of sums of contiguous cells, since we are targeting range queries. One of the best ways to achieve such a result is to calculate an ordinary least squared estimator (OLS) for the counts, as Cormode did in [2]. Generally, such computation takes time proportional to the cube of the number of cells $O(|D|^3)$, thus it is very expensive. Since Cormode considered partitionings with fixed fanout and a rigid privacy budget scheme he managed to exploit these regularities to obtain an estimator in time $O(|D|)$.

Going back to our case, while adopting existing well-studied partitioning schemes allows us to exploit some of their properties, on the other hand the resulting tree is likely to be quite irregular. Producing unbalanced trees with varying budgets makes it difcult if not impossible to compute an OLS in decent time. Also, the deepest branch in the tree would determine the total privacy budget of the algorithm.

This means we could as well release a complete tree with all branches at the deepest level and pay the same budget, thus providing analysts with a much more detailed information about the points distribution. For these reasons, using elaborated partitioning schemes do not appear to be a very competitive choice against existing differentially private solutions. An ad-hoc scheme would be required, such as the one given by the recent DP-Tree method.

5.2 Making Up Generalization-based Algorithms For DP

Very recently, Qardaji et al. [15] in an extended abstract, follow on the work from Li et al.[13] and propose a general framework coined RPS (Recursive Partitioning and Summarization) to achieve DP b.t.w. of multidimensional partitioning. Basically, an RPS algorithm species three subroutines

1. How a region can be partitioned

2. When to stop partitioning

3. How to summarize records in partition

For an RPS framework to be differentially private, all the three subroutines must follow some form of DP. Proposal from Li et al. [13] can then be seen as one way to instantiate that generic RPS framework. It relies on a 2 step process:

1. Random sampling

2. Exponential mechanism to achieve DP multi dimensional partitioning.

The major drawback of the second step is the tractability of the exponential mechanism. Indeed, it requires to virtually generate and rank all the grid resolutions in a way that it becomes possible to order any pair of grids on the given dataset (w.r.t. a given utility function).

To overcome this limitation yet recycling generalization algorithms, we then propose to switch to a DP relaxation paradigm.

5.3 The BangA Case study

Following Gehrke et al. [7] DP relaxation, we adopt the following framework for achieving (sort of) DP:

1. Random sampling

2. Regular generalization mechanism with multidimensional partitioning

3. Laplace mechanism for blocks with small counts BangA has been proved to manage sanitization by generalization mechanism in a very efficient way and with a high quality output. Then we legitimately propose to settle a sanitization process to achieve crowd blending with BangA at the heart.

The remaining work consists in evaluating different settings for the random sampling and their impact on the utility of the output. Optimizations for reducing the requirement of noisy counts in the third step should also be addressed.

6. Conclusion

We show in this communication that there exists a recent and appealing line of research where sophisticated partitioning techniques, K-anonymity algorithms and semantic privacy models can meet and join to the fast and reliable sanitization mechanisms for privacy preserving data publishing. We draw the very first attempt to set up an instantiation of that promising research track b.t.w. of the BANG le, BangA, the crowd blending privacy and random sampling.

References

[1] Anjum, A., Raschia, G. (2011). Spatial indexing combined with clustering for privacy-preserving data publishing. Actes des

journ–ees Bases de Donnees Avancees (BDA'2011) Rabat, Morocco.

[2] Graham Cormode. (2011). Personal privacy vs population privacy: learning to attack anonymization. *In*: Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining, KDD '11, p. 1253{1261, New York, NY, USA, 2011. ACM.

[3] Dwork, C. (2010). Dierential privacy. Automata, languages and programming, p. 1. 12, 2006.

[4] Dwork, C., Smith, A. Dierential privacy for statistics: What we know and what we want to learn. *Journal of Privacy and Confidentiality*, 1 (2) 2.

[5] Cynthia Dwork., Krishnaram Kenthapadi., Frank McSherry., Ilya Mironov., Moni Naor. (2006). Our data, ourselves: Privacy via distributed noise generation. *In*: Advances in Cryptology-EUROCRYPT, p 486-503. Springer.

[6] Freeston, M. (2012). Advances in the design of the BANG le. Foundations of Data Organization and Algorithms, p. 321-338.

[7] Gehrke, J., Hay, M., Lui, E., Pass. R. (2012). Crowd-blending privacy. Advances in Cryptology{CRYPTO 2012, p. 479-496.

[8] Arpita Ghosh., Tim Roughgarden., Mukund Sundararajan. (2012). Universally utility-maximizing privacy mechanisms. SIAM Journal on Computing, 41 (6) 1673-1693.

[9] Iwuchukwu, T., Naughton, J. F. (2007). K-anonymization as spatial indexing: Toward scalable and incremental anonymization. *In*: Proceedings of the 33rd International Conference on Very large data bases, p 746{757. VLDB Endowment, 2007.

[10] Kifer, D., Gehrke, J. (2006) Injecting utility into anonymized datasets. *In*: Proceedings of the 2006 ACM SIGMOD international conference on Management of data, p. 228. ACM.

[11] LeFevre, K., DJ DeWitt., Ramakrishnan, R. (2006). Mondrian multidimensional k-anonymity. *In*: Data Engineering, 2006. ICDE'06. *In*: Proceedings of the 22nd International Conference on, p. 25.

[12] Chao, Li., Michael Hay., Vibhor Rastogi., Gerome Miklau., Andrew McGregor. (2010). Optimizing linear counting queries under differential privacy. In Proceedings of the twenty-ninth ACM SIGMOD SIGACT-SIGART symposium on Principles of database systems, p. 123-134. ACM.

[13] Li, N., Qardaji, W. H., Su. D. (2011). Provably private data anonymization: Or, k-anonymity meets dierential privacy. CoRR,

abs/1101.2604.

[14] Frank McSherry., Kunal Talwar. (2007). Mechanism design via differential privacy. In Foundations of Computer Science, 2007. FOCS'07. 48th Annual IEEE Symposium on, p. 94-103. IEEE.

[15] Qardaji, W., L i. (2012). Recursive partitioning and summarization: a practical framework for dierentially private data publishing. *In*: Proceedings of the 7th ACM Symposium on Information, Computer and Communications Security, p. 38-39. ACM.

[16] Samet. H. (2006). Foundations of multidimensional and metric data structures. Morgan Kaufmann, 2006.

[17] Sweeney. L. (2002). Achieving k-anonymity privacy protection using generalization and suppression. *International Journal of Uncertainty Fuzziness and Knowledge-Based Systems*, 10 (5) 571-588.

[18] Sweeney. L. (2002). k-anonymity: A model for protecting privacy. *International Journal of Uncertainty Fuzziness and Knowledge Based Systems*, 10 (5) 557-570.

[19] Wei Wang., Jiong Yang., Richard Muntz. (2000). *Information organization and databases*. chapter PK-tree: a spatial index structure for high dimensional point data, p. 281-293. Kluwer Academic Publishers, Norwell, MA, USA.

[20] Xiao, X., Wang, G., Gehrke. J. (2011). Dierential privacy via wavelet transforms. Knowledge and Data Engineering, IEEE Transactions on, 23 (8) 1200-1214.

[21] Yonghui Xiao., Li Xiong., Chun Yuan. (2010). Dierentially private data release through multidimensional partitioning. *In*: Secure Data Management, p. 150-168. Springer.